

软件(结构)设计说明(SDD)

说明:

- 1.《软件(结构)设计说明》(SDD)描述了计算机软件配置项(CSCI)的设计。它描述了 CSCI 级设计决策、CSCI 体系结构设计(概要设计)和实现该软件所需的详细设计。SDD 可用接口设计说明 IDD 和数据库(顶层)设计说明 DBDD 加以补充。
- 2.SDD 连同相关的 IDD 和 DBDD 是实现该软件的基础。向需方提供了设计的可视性，为软件支持提供了所需要的信息。
- 3.IDD 和 DBDD 是否单独成册抑或与 SDD 合为一份资料视情况繁简而定。

目录

1 引言	3
1.1 标识	3
1.2 系统概述.....	3
1.3 文档概述.....	3
1.4 基线	4
2 引用文件.....	4
3 CSCI 级设计决策.....	4
4 CSCI 体系结构设计.....	5
4.1 体系结构.....	5
4.1.1 程序(模块)划分	5
4.1.2 程序(模块)层次结构关系	9
4.2 全局数据结构说明.....	10
4.3 CSCI 部件.....	12
4.4 执行概念.....	15
执行概念.....	15
4.5 接口设计.....	20
4.5.1 接口标识与接口图.....	20
5 CSCI 详细设计.....	23
6 需求的可追踪性.....	26
7 注解	27

1 引言

1.1 标识

本条应包含本文档适用的系统和软件的完整标识。(若适用)包括标识号、标题、缩略词语、版本号、发行号。

1.2 系统概述

开发团队：无敌暴龙兽

成员学号	姓名	技术特点	团队角色
202100130026	田凯峰	前端	项目经理
202100130213	刘喜	后端	后端
202100130209	粟锦虹	前端,算法	前端
202100130196	李华焄	算法	算法
202017190218	高子寒	测试,运营	测试,运营

1.3 文档概述

用途与内容：

本软件结构设计说明 (SAD) 文档详细定义了“人事管理系统”项目的软件结构。本文档旨在为开发人员、测试人员、提供对系统结构、性能和接口的全面理解。该文档涵盖了系统的系统架构，数据结构说明和外部接口等方面，提供系统设计的详细信息和指导，帮助团队成员理解和实现软件系统，同时为系统的维护和扩展提供支持。

保密性与私密性要求：

本文档包含敏感信息，例如系统设计、数据库结构和业务流程。因此，该文档仅供授权人员访问，包括：

项目开发团队成员

项目管理人员

未经授权，不得复制、分发或披露本文件中的任何信息。所有阅读或使用本文件的人员都应遵守相关的保密协议和数据保护法规。

1.4 基线

说明编写本系统设计说明书所依据的设计基线。

1. CSCI 级设计决策
2. CSCI 级体系结构设计
3. CSCI 级详细设计
4. 需求可追踪性

2 引用文件

暂无

3 CSCI 级设计决策

人事管理系统的 CSCI 级设计决策

a. 输入和输出的设计决策

输入接口用户通过系统界面输入员工、部门和薪资管理等信息。

用户接口设计直观，使用标准的 Web 表单元素用于数据录入，例如文本框、下拉菜单和按钮。

输出接口系统生成员工、部门和薪资报告，提供网页显示。

实时数据显示，包括员工状态和薪资统计图表（如饼图、柱状图）。

对于接口的具体技术细节，可以参考系统的接口设计说明（IDD）。

b. 行为设计决策

系统行为

系统根据用户输入的数据自动更新对应的员工信息、部门信息和薪资信息。

每次修改数据后实时更新显示，无需用户手动刷新页面。

针对常见输入错误（如数据格式不正确）提供即时反馈和错误提示。

响应时间

用户操作后的响应时间控制在 2 秒以内，以保证系统的良好用户体验。

数据处理和统计操作在后台执行，前端仅显示结果。

容错处理

对于不允许的输入（如负数薪资、无效日期）实时进行校验和提示。

对于系统异常（如数据库连接失败）提供友好的错误提示并记录日志以供后台分析。

c. 数据库/数据文件展示设计决策

数据展示

系统界面设计简洁明了，通过表格、列表和图表展示员工、部门和薪资数据。

提供分页和搜索功能，以使用户快速查找所需信息。

用户权限

不同用户角色（如管理员、普通员工）根据权限访问不同的数据。

管理员可以查看和管理所有数据，其他用户根据权限查看或编辑相关数据。

有关数据库设计的详细信息请参见数据库设计说明（DBDD）。

d. 满足安全性、保密性、私密性的设计决策

数据加密

敏感数据（如用户密码）在数据库中进行加密存储。

访问数据根据令牌技术来拦截和过滤

用户认证和授权

使用令牌技术（JWT）进行用户认证，确保用户身份的合法性。

基于角色的访问控制（RBAC），不同角色具有不同的权限。

数据备份和恢复

定期自动备份数据库，保证数据丢失后能够快速恢复。

定期检查存储数据库的稳定性

使用日志记录操作，便于恢复数据

e. 其他设计决策

系统灵活性

系统设计模块化，便于功能扩展和维护。

支持插件机制，未来可以方便地添加新功能模块。

前后端分离设计，开发更加方便

可用性

界面设计符合用户习惯，提高操作的直观性和效率。

提供丰富的帮助文档和用户指南，帮助用户快速上手系统。

可维护性

代码遵循良好的编码规范和注释规范，便于后期维护和升级。

使用版本控制系统管理代码，记录每次修改历史。

4 CSCI 体系结构设计

本章应分条描述 CSCI 体系结构设计。如果设计的部分或全部依赖于系统状态或方式，则应指出这种依赖性。如果设计信息在多条中出现，则可只描述一次，而在其他条引用。应给出或引用为理解这些设计所需的设计约定。

4.1 体系结构

4.1.1 程序(模块)划分

以下是人事管理系统的程序模块划分和其功能描述：

UI 模块：主要为网页前端部分

名称: `UserInterface`

标识符: `UI_01`

功能: 提供用户界面，处理用户输入和输出显示，包含登录、员工管理、部门管理、薪资管理和数据统计页面。

源标准名: 部分截图如下

```
export const constantRoutes = [
  {
    path: "/login",
    component: () => import("@/views/login/index"),
    hidden: true,
  },
  {
    path: "/register",
    component: () => import("@/views/register/index"),
    hidden: true,
  },
  {
    path: "/404",
    component: () => import("@/views/404"),
    hidden: true,
  },
  {
    path: "/",
    component: Layout,
    redirect: "/dashboard",
    children: [
      {
        path: "dashboard",
        name: "Dashboard",
        component: () => import("@/views/dashboard/index"),
      }
    ]
  }
]
```

```

component: Layout,
redirect: "/system",
name: "System",
meta: { title: "系统信息管理", icon: "el-icon-s-tools" },
children: [
  {
    path: "dept",
    name: "Dept",
    component: () => import("@/views/dept"),
    meta: { title: "部门管理", icon: "el-icon-menu" },
  },
  {
    path: "emp",
    name: "Emp",
    component: () => import("@/views/emp"),
    meta: { title: "员工管理", icon: "el-icon-user-solid" },
  },
],
},
{
  path: "/report",
  component: Layout,
  redirect: "/report",
  name: "Report",
  meta: { title: "数据统计管理", icon: "el-icon-s-tools" },
  children: [

```

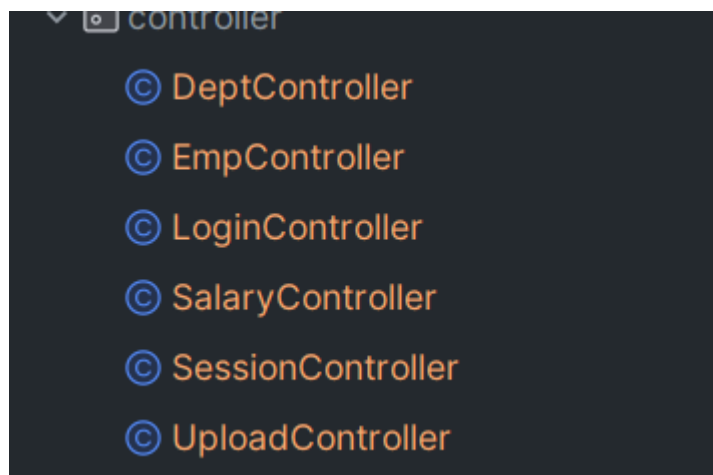
业务逻辑模块

名称: BusinessLogic

标识符: BL_01

功能: 处理所有业务逻辑，接口提供给 UI 模块使用，包含员工管理逻辑、部门管理逻辑、薪资管理逻辑和统计逻辑。

源标准名:



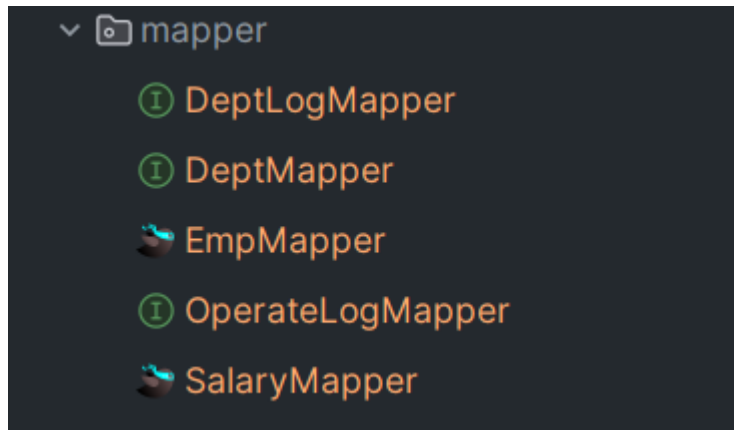
数据库访问模块

名称: DataAccess

标识符: DA_01

功能: 负责与数据库的交互, 执行 CRUD 操作, 包含员工数据管理、部门数据管理和薪资数据管理。

源标准名: Mapper 层访问数据库, 直接与数据库操作



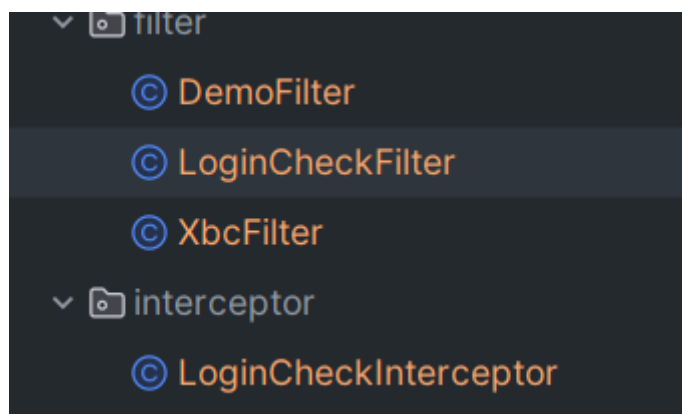
安全模块

名称: Security

标识符: SEC_01

功能: 负责用户认证和权限管理, 以及数据加密和解密, 包含认证服务、授权服务和加密服务。

源标准名:



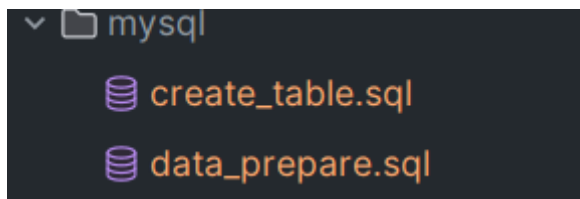
数据模型模块

名称: DataModel

标识符: DM_01

功能: 定义系统中使用的数据结构和数据模型, 包含员工、部门和薪资的模型定义。

源标准名:



```
spring:
  #数据库连接信息
  datasource:
    driver-class-name: com.mysql.cj.jdbc.Driver
    url: jdbc:mysql://localhost:3306/personnelmanagement
    username: root
    password: 
  #文件上传的配置
  servlet:
    multipart:
      max-file-size: 10MB
      max-request-size: 100MB
  # 端口设置
  server:
    port: 9000
```

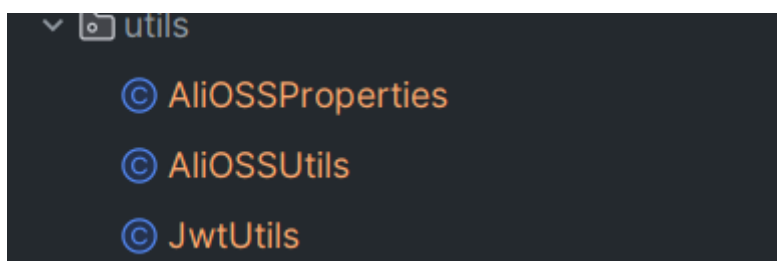
公用工具模块

名称: Utility

标识符: UT_01

功能: 提供系统中使用的通用工具，例如日志记录、文件处理和配置管理。

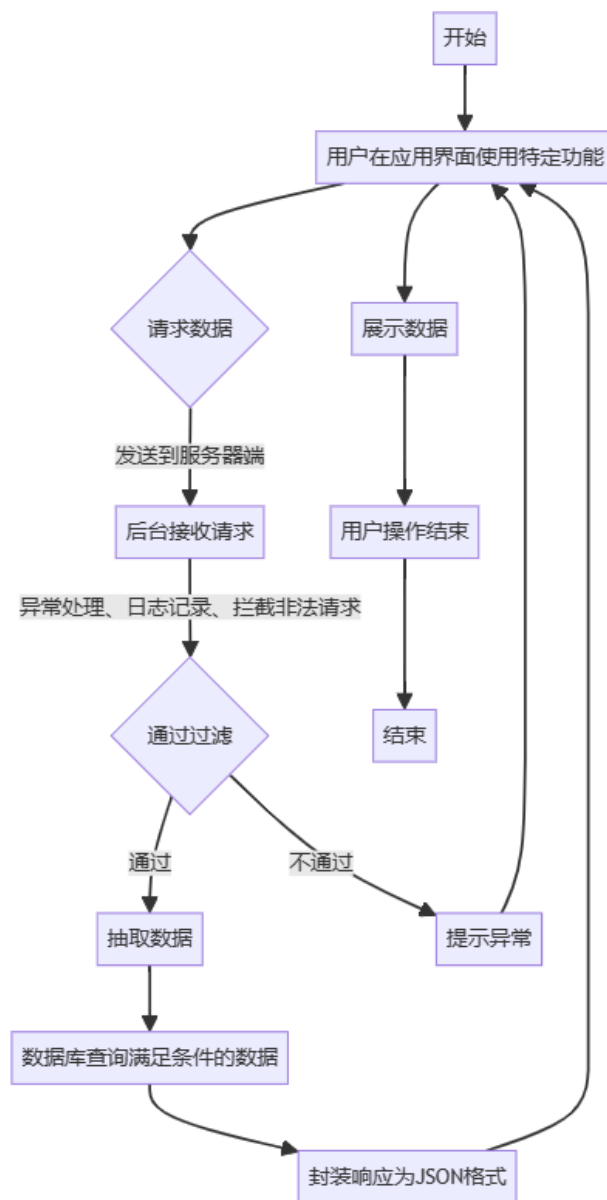
源标准名:



4.1.2 程序(模块)层次结构关系

用一系列图表列出本 CSCI 内的每个程序(包括每个模块和子程序)之间的层次结构与调用关

系。



4.2 全局数据结构说明

本章说明本程序系统中使用的全局数据常量、变量和数据结构。

1. 员工信息数据

数据描述：员工管理系统需要包含员工的详细信息

数据元素：（列出部分，完整信息在描述约定中给出每个数据元素的格式大小和设置等）

姓名：字符串类型，最大长度 30 个字符

入职日期（离职日期等日期信息）：日期类型，精确到天即可

职位：字符串类型，最大长度 20 个字符

工资：数值类型，保留两位小数

性别：小整型，用 0,1 标识性别

身份证号码：字符串，固定为 18 位（暂未考虑特殊情况）

... 省略

数据存储：存储在员工信息表中

数据约束：员工 ID 必须是唯一的，工资不能为负数

2. 部门信息数据

数据描述：系统需要维护部门的相关信息

数据元素：

```
id int unsigned (auto increment)
name varchar(10)
create_time datetime
update_time datetime
```

部门名称：字符串类型，最大长度 10 个字符

部门 ID：无符号整型，部门 ID 自增

部门成立日期：日期类型，为部分创建时间

数据存储：存储在部门信息表中

数据约束：部门名称不得重复

此外，系统还记录了每次操作的日志

3. 操作日志数据

数据描述：操作日志表用于记录系统的操作历史，包括操作人和操作描述。

数据元素：

```
operate_log
列 8
id int unsigned (auto increment)
operate_user int unsigned
operate_time datetime
class_name varchar(100)
method_name varchar(100)
method_params varchar(1000)
return_value varchar(2000)
cost_time bigint
```

操作时间：日期时间类型

操作人 ID：无符号 int 类型，为操作人 ID

操作描述：文本类型，记录操作的详细描述

...其余如图

数据存储：存储在操作日志表中

数据约束：操作时间不能为 NULL，操作人不能为空，操作描述必须提供

4.3 CSCI 部件

本条应：

a.标识构成该 CSCI 的所有软件配置项。应赋予每个软件配置项一个项目唯一标识符。

注:软件配置项是 CSCI 设计中的一个元素,如 CSCI 的一个主要的分支、该分支的一个组成部分、一个类、对象、模块、函数、例程或数据库.软件配置项可以出现在一个层次结构的不同层次上,并且可以由其他软件配置项组成.设计中的软件配置项与实现它们的代码和数据实体(例程、过程、数据库、数据文件等)或包含这些实体的计算机文件之间,可以有也可以没有一对一的关系。一个数据库可以被处理为一个 CSCI,也可被处理为一个软件配置项。SDD 可以通过与所采用的设计方法学一致的名字来引用软件配置项。

b.给出软件配置项的静态关系(如“组成”)。根据所选择的软件设计方法学可以给出多种关系(例如,采用面向对象的设计方法时,本条既可以给出类和对象结构,也可以给出 CSCI 的模块和过程结构)。

c.陈述每个软件配置项的用途,并标识分配给它的 CSCI 需求与 CSCI 级设计决策(需求的分配也可在 6.a 中提供)。

d.标识每个软件配置项的开发状态/类型(如新开发的软件配置项、重用已有设计或软件的软件配置项、再工程的已有设计或软件、为重用而开发的软件等)。对于已有设计或软件,本说明应提供标识信息,如名称、版本、文档引用、库等。

e.描述 CSCI(若适用,每个软件配置项)计划使用的计算机硬件资源(例如处理器能力、内存容量、输入/输出设备能力、辅存容量和通信/网络设备能力)。这些描述应覆盖该 CSCI 的资源使用需求中提及的、影响该 csci 的系统级资源分配中提及的、以及在软件开发计划的资源使用度量计划中提及的所有计算机硬件资源。如果一给定的计算机硬件资源的所有使用数据出现在同一个地方,如在一个 SDD 中,则本条可以引用它。针对每一计算机硬件资源应包括如下信息:

- 1)得到满足的 CSCI 需求或系统级资源分配;
- 2)使用数据所基于的假设和条件(例如,典型用法、最坏情况用法、特定事件的假设);
- 3)影响使用的特殊考虑(例如虚存的使用、覆盖的使用、多处理器的使用或操作系统开销、库软件或其他的实现开销的影响);
- 4)所使用的度量单位(例如处理器能力百分比、每秒周期、内存字节数、每秒千字节);
- 5)进行评估或度量的级别(例如软件配置项,CSCI 或可执行程序)。

f.指出实现每个软件配置项的软件放置在哪个程序库中。

如下:

a. 标识构成该 CSCI 的软件配置项

以下是软件配置项及其唯一标识符:

UserInterface (UI) - CSCI_UI_001

BusinessLogic (BL) - CSCI_BL_002

DataAccess (DA) - CSCI_DA_003

Security (SEC) - CSCI_SEC_004

DataModel (DM) - CSCI_DM_005

Utility (UTL) - CSCI_UTL_006

b. 软件配置项的静态关系

下面给出这些软件配置项的组成关系：



c. 每个软件配置项的用途及其分配的需求和设计决策

UserInterface (CSCI_UI_001)

用途: 提供用户与系统交互的界面。

需求和设计决策: 用户输入处理、界面呈现、用户交互响应。

开发状态: 新开发并待完善。

BusinessLogic (CSCI_BL_002)

用途: 执行系统的核心业务逻辑。

需求和设计决策: 业务规则处理、任务调度。

开发状态: 根据需求进行新开发。

DataAccess (CSCI_DA_003)

用途: 负责数据的访问和管理。

需求和设计决策: 数据库连接管理、数据查询和更新。

开发状态: 重用已有的数据访问模块。

Security (CSCI_SEC_004)

用途: 确保系统的安全性，包括认证和授权。

需求和设计决策: 用户认证、权限管理、安全审计。

开发状态: 重用现有的安全组件。

DataModel (CSCI_DM_005)

用途: 定义系统的数据结构和数据模型。

需求和设计决策: 数据结构定义、数据验证。

开发状态: 新开发，并部分重用已有数据模型。

Utility (CSCI_UTL_006)

用途: 提供通用的工具函数和实用程序。

需求和设计决策: 日志记录、数据转换、辅助工具，图片上传等。

开发状态: 已开发，待完善（如身份访问控制 RABC 完善）。

d. 软件配置项的开发状态/类型

UserInterface (CSCI_UI_001): 新开发, 部分功能待完善
BusinessLogic (CSCI_BL_002): 完善功能, 与前端联调中
DataAccess (CSCI_DA_003): 重用已有模块
Security (CSCI_SEC_004): 重用现有组件
DataModel (CSCI_DM_005): 新开发和部分重用 (数据库 MySQL)
Utility (CSCI_UTL_006): 使用已有工具, 暂未新增工具类

e. 计算机硬件资源描述

得到满足的需求/系统级资源分配

UserInterface: 高分辨率显示器, 快速响应的输入设备。

BusinessLogic: 高性能处理器、多核支持。

DataAccess: 大内存容量、高 IO 带宽。

Security: 专用加密硬件模块。

DataModel: MySQL 数据库, 大存储容量, 用于存放数据文件。

Utility: 高性能计算资源, 云端存储图片文件。

使用数据所基于的假设和条件

典型用法: 正常操作条件下的资源需求分析。

最坏情况用法: 高峰负载下的资源需求分析。

影响使用的特殊考虑

虚存使用: 确保内存管理的高效性。

多处理器使用: 任务分发和并行处理。

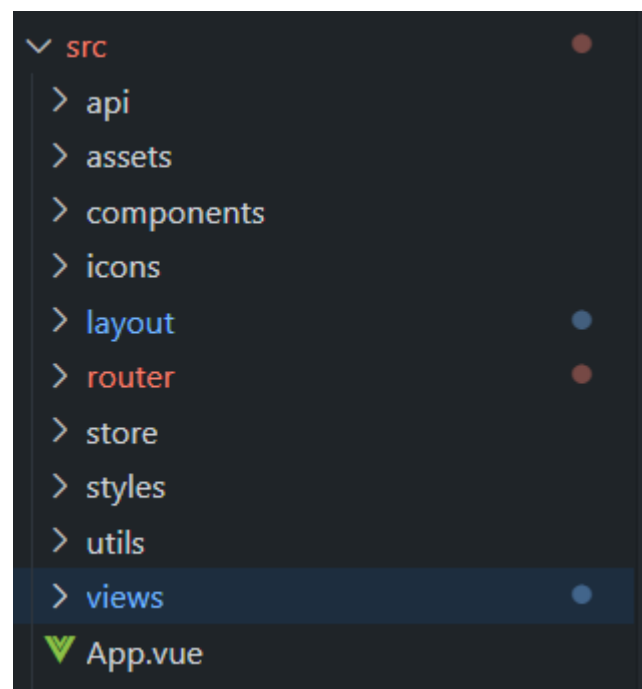
所使用的度量单位

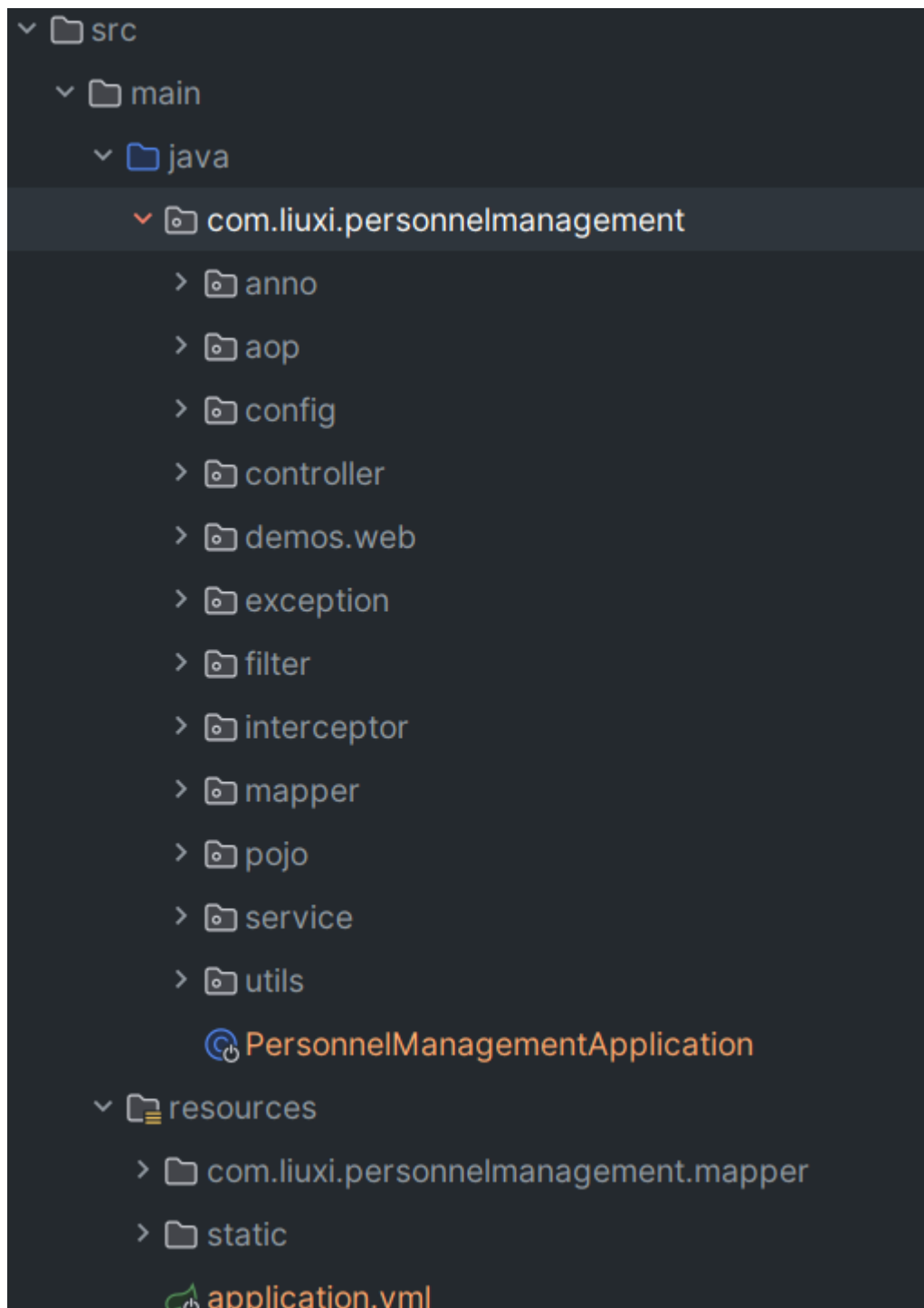
处理器能力百分比、内存字节数、每秒 IO 操作数。

进行评估或度量的级别

评估级别: CSCI 总评估, 配置项级评估。

f. 软件配置项存放库





4.4 执行概念

执行概念

为了描述软件配置项（CSCI）之间的执行概念，我们采用图示和说明来解释它们在运行期间如何交互以及各个模块的动态行为。此包括执行控制流、数据流、动态控制序列、状态转换

图、时序图、配置项之间的优先关系、中断处理、异常处理、并发执行等。

名词解释：

UserInterface： 用户接口，即前端 UI

BusinessLogic： 业务逻辑，即部门管理，员工管理，数据分析业务

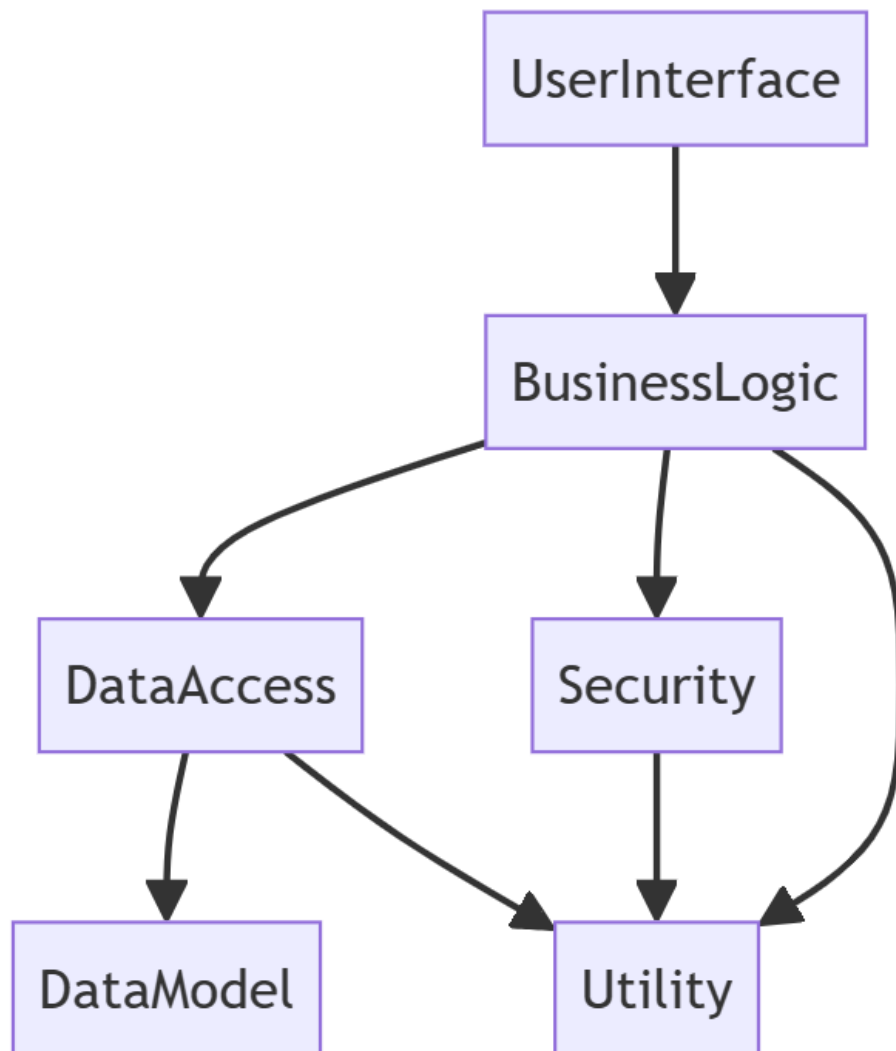
DataAccess： 数据操作

Security： 安全性验证等方面

DataModel and Utility： 数据模型和工具类

1. 执行控制流

下图描述了各模块之间的执行控制流：

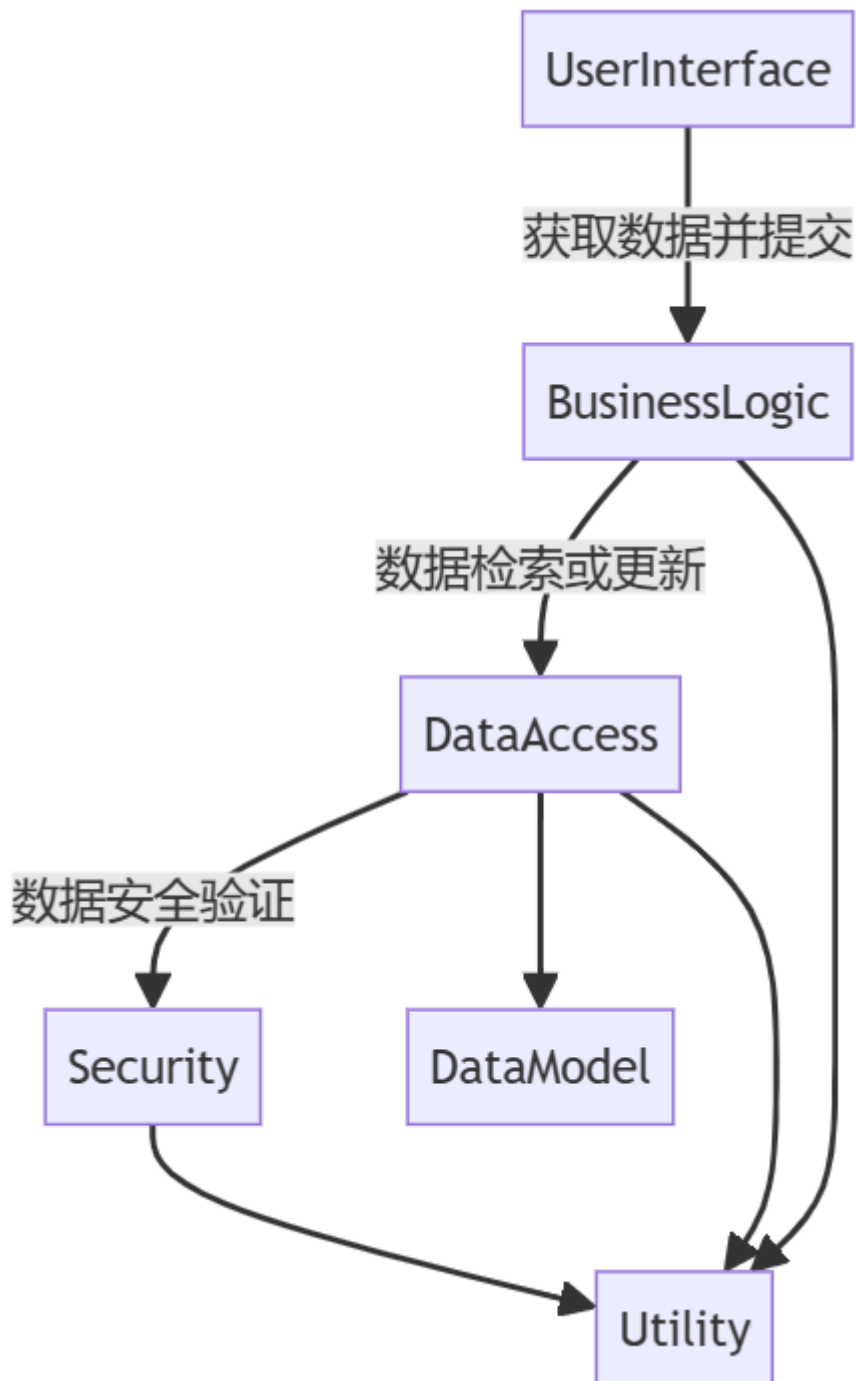


说明：

用户操作由 `UserInterface` 模块发起，之后控制流传递到 `BusinessLogic` 模块。
`BusinessLogic` 模块根据需要调用 `DataAccess` 和 `Security` 模块以完成具体的业务逻辑任务。
`DataAccess` 模块负责与数据库交互，而其内部可能会调用 `DataModel` 模块用于数据的结构化处理。
`Utility` 模块提供通用服务。

2. 数据流

数据流描述了数据在模块之间的流动：



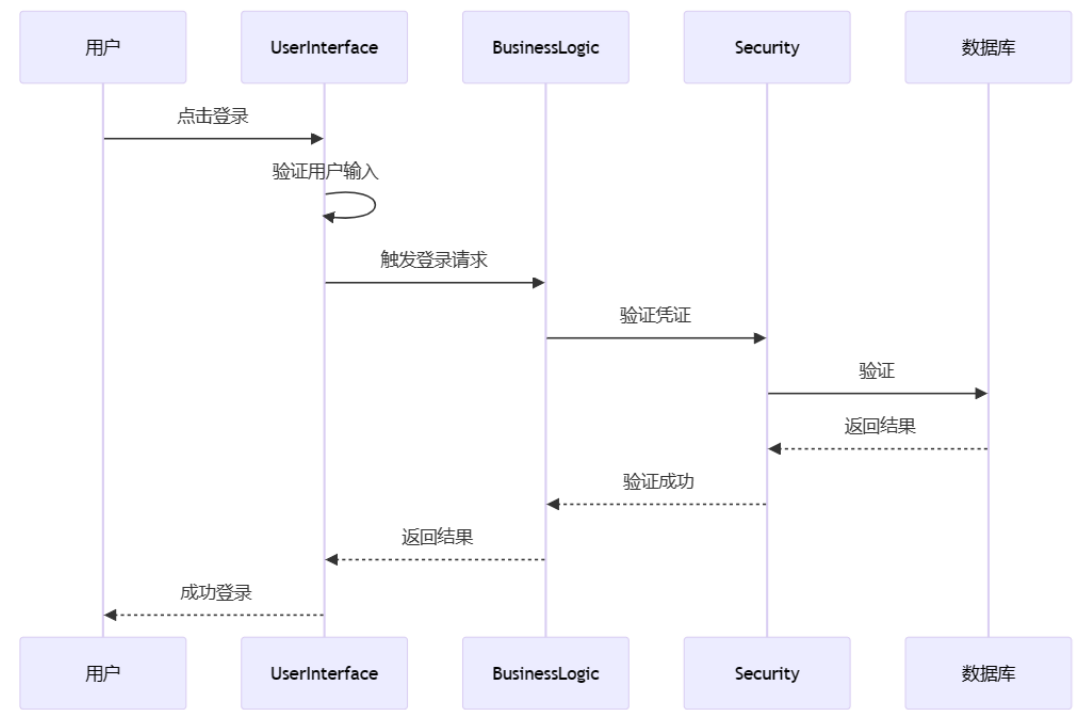
说明：

`UserInterface` 模块从用户处获取数据并提交到 `BusinessLogic` 模块。
`BusinessLogic` 模块根据业务逻辑，调用 `DataAccess` 模块进行数据检索或更新。

DataAccess 模块在操作数据库前，可能调用 Security 模块进行数据安全验证。
 DataModel 模块用于定义数据的结构和格式。

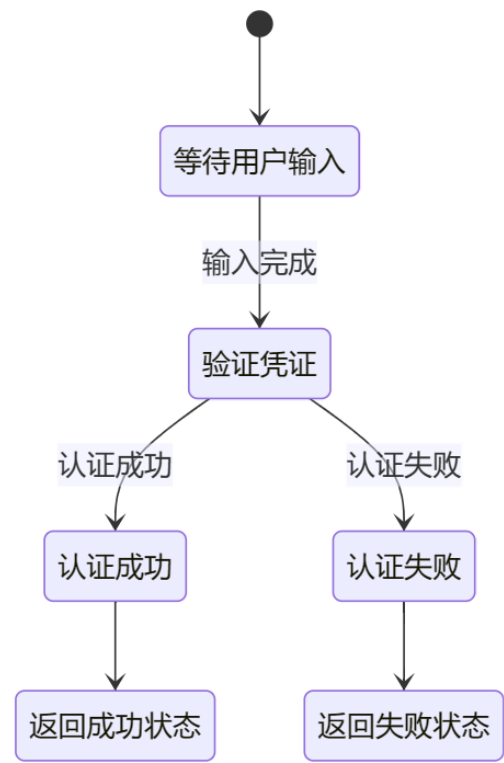
3. 动态控制序列与时序图

以下时序图展示了用户登录的示例：



4. 状态转换图

以下是用户认证（登录）模块的状态转换图：

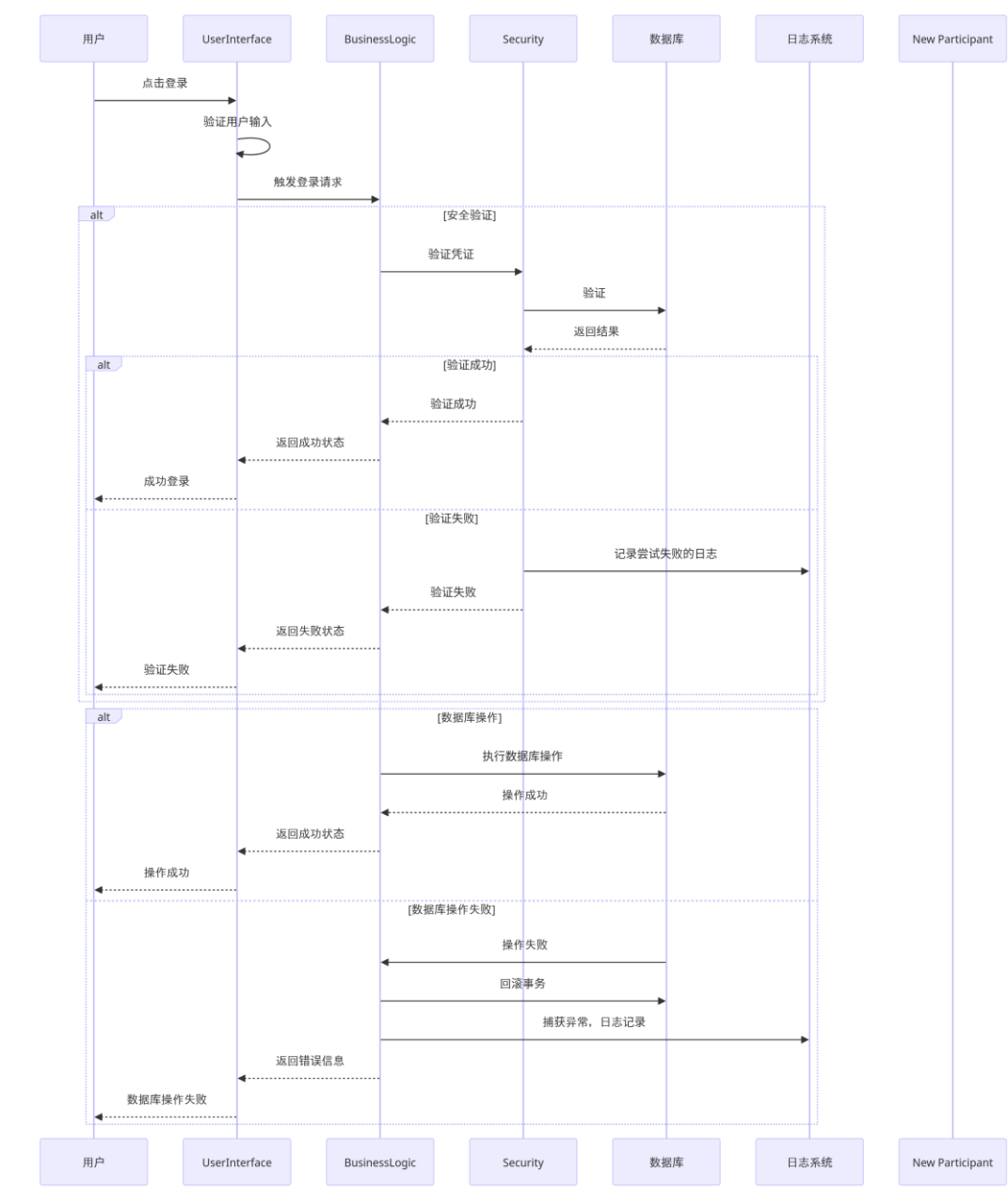


5. 异常处理

异常处理：

数据库操作失败：捕获异常，回滚事务，日志记录，并向用户返回错误信息。

安全验证失败：记录尝试失败的日志，返回错误告知用户。



6. 动态分配与去分配

在运行期间，系统根据需求动态分配和释放资源，例如数据库连接、内存对象等。以下是一个示例解释动态分配与去分配：

示例：在请求数据库连接时，DataAccess 模块会从连接池中获取一个连接，完成操作后将连接返回到池中进行回收。

数据库连接池：本项目使用 Hikari 数据库连接池接口，同时保留德鲁伊数据库连接池接口：

切换Druid数据库连接池

官方地址: <https://github.com/alibaba/druid/tree/master/druid-spring-boot-starter>

```
<dependency>
  <groupId>com.alibaba</groupId>
  <artifactId>druid-spring-boot-starter</artifactId>
  <version>1.2.8</version>
</dependency>
```

引入依赖即可使用

pom.xml

7. 中断处理、异常处理

中断处理在现代应用程序中通常由操作系统管理,例如来自硬件或软件的中断。对于应用层,只需要处理异常情况。例如:

数据库超时(软中断处理): 超时后,抛出异常并记录日志,尝试重试或返回错误消息。

```
// @Transactional 开启事务管理—spring框架,并且指定所有异常都回滚
1 个用法
@Transactional(rollbackFor = Exception.class)
@Override
public void delete(Integer id) {
    try
    {
        // 删除部门
        deptMapper.deleteById(id);
        // 部门下的员工也会被删除
        empMapper.deleteByDeptId(id);
    }
}
```

4.5 接口设计

4.5.1 接口标识与接口图

a. 用户接口

1. 项目唯一标识符: UMS-UI-01

名称/标识符: 用户登录界面

数据类型: 文本输入

大小和格式：用户名（最多 20 个字符），密码（最少 6 个字符，最多 20 个字符）

```
username varchar(20)
password varchar(32)
```

准确度和精度：验证用户凭证的准确性

来源/接收者：用户/身份验证模块

优先级别：高

保密性约束：密码加密传

```
import { login } from "@api/user";

data() {
  //用户名校验规则
  const validateUsername = (rule, value, callback) => {
    if (!validUsername(value)) {
      callback(new Error("请输入正确的用户名"));
    } else {
      callback();
    }
  };

  //用户名校验规则
  const validatePassword = (rule, value, callback) => {
    if (value.length < 6) {
      callback(new Error("密码长度至少为6位"));
    } else {
      callback();
    }
  };
}
```

b. 数据管理接口

1) 项目唯一标识符：UMS-DM-01

名称/标识符：数据管理接口

操作类型：增加、删除、编辑、查询员工信息

数据元素：员工记录

准确度和精度：准确地执行数据操作

来源/接收者：员工管理模块/数据库

优先级别：高

安全性/保密性：权限控制要求，根据用户权限来过滤操作

```
import { page, add, update, deleteById, selectById } from "@api/emp.js";
import { findAll } from "@api/dept.js";
```

c. 统计分析接口

1. 项目唯一标识符：UMS-SA-01

名称/标识符：统计分析界面

数据类型：统计数据集

处理：选择统计方法（如总和、平均值、计数）

输出：生成统计图表与报表

优先级别：中

保密性约束：数据保密性要求

内部接口

1. 数据访问接口

接口描述：用于模块间访问数据库的接口

功能：获取、创建、更新和删除员工信息

数据元素：员工记录、部门信息等

优先级：高

2. 认证与授权接口

接口描述：处理用户身份认证和访问控制的接口

功能：验证用户登录信息、控制用户权限

数据元素：用户名、密码、角色信息

优先级：中

3. 数据同步接口

接口描述：用于不同模块间的数据同步

功能：确保数据在各模块间的一致性

数据元素：员工信息、部门信息等

优先级：中

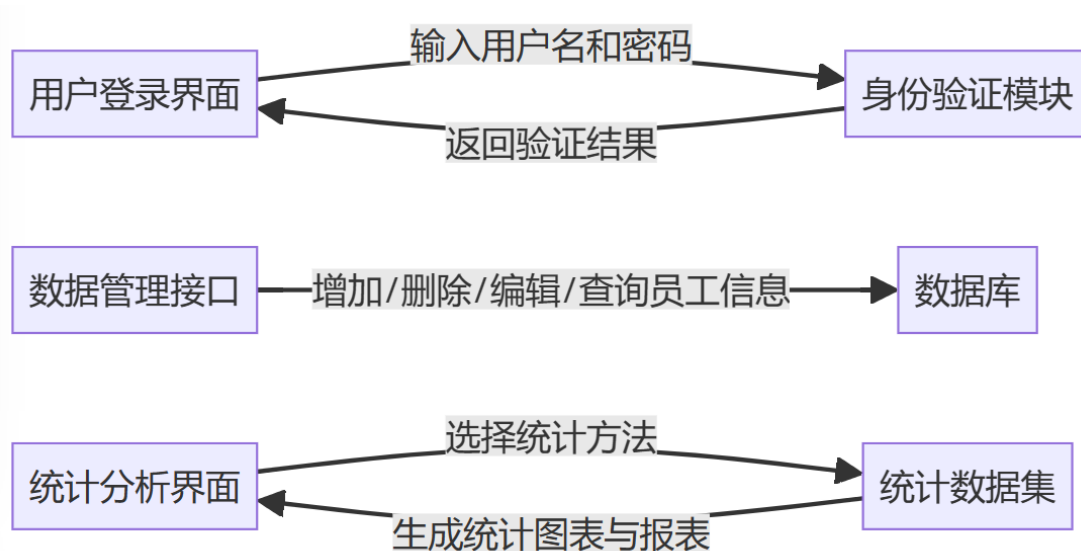
4. 统计分析接口

接口描述：用于生成报表和统计分析

功能：提供数据分析、生成图表和报表

数据元素：统计数据集、图表信息

优先级：低



5 CSCI 详细设计

本章应分条描述 CSCI 的每个软件配置项。如果设计的部分或全部依赖于系统状态或方式，则应指出这种依赖性。如果该设计信息在多条中出现，则可只描述一次，而在其他条引用。应给出或引用为理解这些设计所需的设计约定。软件配置项的接口特性可在此处描述，也可在第 4 章或接口设计说明(IDD)中描述。数据库软件配置项，或用于操作/访问数据库的软件配置项，可在此处描述，也可在数据库(顶层)设计说明(DBDD)中描述。

5.x(软件配置项的项目唯一标识符或软件配置项组的指定符)

本条应用项目唯一标识符标识软件配置项并描述它。(若适用)描述应包括以下信息。作为一种变通，本条也可以指定一组软件配置项，并分条标识和描述它们。包含其他软件配置项的软件配置项可以引用那些软件配置项的说明，而无需在此重复。

- a.(若有)配置项设计决策，诸如(如果以前未选)要使用的算法；
- b.软件配置项设计中的约束、限制或非规范特征；
- c.如果要使用的编程语言不同于该 CSCI 所指定的语言.应该指出，并说明使用它的理由；
- d.如果软件配置项由过程式命令组成或包含过程式命令(如数据库管理系统(DBMS)中用于定义表单与报表的菜单选择、用于数据库访问与操纵的联机 DBMS 查询、用于自动代码生成的图形用户接口(GUI)构造器的输入、操作系统的命令或 shell 脚本)，应有过程式命令列表和解释它们的用户手册或其他文档的引用；
- e.如果软件配置项包含、接收或输出数据，(若适用)应有对其输入、输出和其他数据元素以及数据元素集合体的说明。(若适用)本文的 4.5.x 提供要包含主题的列表。软件配置项的局部数据应与软件配置项的输入或输出数据分开来描述。如果该软件配置项是一个数据库，应引用相应的数据库(顶层)设计说明(DBDD)；接口特性可在此处提供，也可引用本文第 4 章或相应接口设计说明。
- f.如果软件配置项包含逻辑，给出其要使用的逻辑，(若适用)包括：
 - 1)该软件配置项执行启动时，其内部起作用的条件；
 - 2)把控制交给其他软件配置项的条件；
 - 3)对每个输入的响应及响应时间，包括数据转换、重命名和数据传送操作；
 - 4)该软件配置项运行期间的操作序列和动态控制序列，包括：

- a) 序列控制方法;
 - b) 该方法的逻辑与输入条件, 如计时偏差、优先级赋值;
 - c) 数据在内存中的进出;
 - d) 离散输入信号的感知, 以及在软件配置项内中断操作之间的时序关系;
- 5) 异常与错误处理。

5.1 UserInterface (CSCI_UI_001)

- a. 配置项设计决策: 使用基于组件的用户界面设计方法。
- b. 设计约束和特征: 响应式设计, 支持多种屏幕尺寸。
- c. 编程语言: 基于 Vue 框架, 使用 HTML、CSS 和 JavaScript 和 element, 以满足跨平台要求。
- d. 过程式命令: 用户界面交互过程/工作流程。
- e. 数据输入/输出: 用户输入表单数据, 显示结果页面。
- f. 逻辑设计:

启动条件: 用户请求特定功能。

控制转移条件: 用户导航至其他功能模块。

输入响应: 数据验证、处理、显示。

运行期间操作: 页面加载顺序、事件处理逻辑。

异常处理: 错误信息显示、重试机制。

5.2 BusinessLogic (CSCI_BL_002)

- a. 配置项设计决策: 使用基于业务规则引擎的逻辑执行方法。
- b. 设计约束和特征: 实现快速决策逻辑, 提高系统效率。
- c. 编程语言: Java 基于 springboot2 开发 JavaWeb 项目, 为性能和扩展性所选。
- d. 过程式命令: 业务流程处理指令列表。
- e. 数据输入/输出: 接收来自 UI 的请求, 处理后返回结果。
- f. 逻辑设计:

启动条件: 收到业务请求。

控制转移条件: 数据传递至数据访问或安全模块。

输入响应: 数据转换、业务规则验证。

运行期间操作: 安全性验证、数据操作逻辑。

异常处理: 异常情况处理、日志记录。

5.3 DataAccess (CSCI_DA_003)

- a. 配置项设计决策: 使用对象关系映射 (ORM) 框架进行数据操作, 使用 MySQL 数据库。
- b. 设计约束和特征: 数据库连接池管理、查询性能优化。
- c. 编程语言: SQL 语言, 使用 MyBatis 优化 SQL 且功能更强大, 使用适合与 ORM 库集成。
- d. 过程式命令: 数据库查询语句列表和连接管理指令。
- e. 数据输入/输出: 从业务逻辑模块接收请求, 执行数据库操作, 返回结果。
- f. 逻辑设计:

启动条件: 收到数据访问请求。

控制转移条件: 数据查询结果返回给业务逻辑。

输入响应: 数据库操作响应包括读写操作。

运行期间操作: SQL 查询执行、连接管理。

异常处理: 数据库错误处理、事务控制逻辑。

5.4 安全性 (Security, SEC) - CSCI_SEC_004

- a. 配置项设计决策:

采用 JWT 令牌进行用户身份验证和授权。

使用加密算法（如 AES-256 算法）进行数据加密（暂代完善）。

b. 设计约束和特征:

必须符合 GDPR 等相关数据保护法规。

高安全性的需求导致性能可能受到一定的影响。

c. 编程语言: 使用 Java 实现。

d. 过程式命令: 安全策略设置命令和密钥管理指令。

e. 数据输入/输出:

接收用户认证请求, 验证用户凭证, 生成访问令牌并返回给客户端。

处理加密和解密的数据请求。

f. 逻辑设计:

启动条件: 接收到用户认证或授权请求。

控制转移条件: 身份验证成功后将控制权交给业务逻辑模块。

输入响应: 对请求的验证和数据加密操作。

运行期间操作: 认证、授权、加密和解密操作。

异常处理: 用户身份验证失败处理、加密解密错误处理、访问令牌过期处理。

5.5 数据模型 (DataModel, DM) - CSCI_DM_005

a. 配置项设计决策:

使用实体关系模型 (ERM) 定义数据结构。

选择 MySQL 作为数据库管理系统。

b. 设计约束和特征:

数据库规范化以减少冗余, 提高数据完整性。

考虑数据库扩展性和性能优化。

c. 编程语言: 使用 SQL 定义数据库架构。

d. 过程式命令: 数据定义语言 (DDL) 语句列表, 用于创建和修改数据库结构。

e. 数据输入/输出: 数据由数据访问模块输入和输出, 包括插入、更新、删除和查询操作。

f. 逻辑设计:

启动条件: 初始化或更新数据库结构时启动。

控制转移条件: 数据操作请求传递到数据访问模块。

输入响应: 处理数据库操作请求, 如插入新记录或更新已有记录。

运行期间操作: 管理数据库表和关系。

异常处理: 数据完整性失败处理、数据类型转换错误处理。

5.6 实用工具 (Utility, UTL) - CSCI_UTL_006

a. 配置项设计决策:

提供日志记录、调试和性能监控工具等。

选择 Log4j 作为日志记录框架。

b. 设计约束和特征:

日志记录需要与应用程序的其他模块无缝集成。

工具应具有高效的性能, 不影响主要功能的运行。

c. 编程语言: 使用 Java 实现日志记录和调试工具。

d. 过程式命令: 日志记录命令、系统监控和调试指令。

e. 数据输入/输出:

记录输入: 应用程序中的操作日志和调试信息。

输出: 生成日志文件, 提供调试和监控数据。

- f. 逻辑设计:
- 启动条件: 应用程序启动或发生特定事件时触发。
- 控制转移条件: 记录完成后将控制权返回调用模块。
- 输入响应: 处理并记录日志信息, 执行调试与监控操作。
- 运行期间操作: 实时日志记录、调试信息捕获、性能监控。
- 异常处理: 日志记录失败处理、监控信息收集失败处理。

6 需求的可追踪性

- 本章应包括:
- a.从本 SDD 中标识的每个软件配置项到分配给它的 CSCI 需求的可追踪性(亦可在 4.1 中提供);
 - b.从每个 CSCI 需求到它被分配给软件配置项的可追踪性。

对需求的追踪主要通过人工追踪的方式进行完成, 通过准备一个 EXCEL 表格, 其中罗列人事管理系统所需要的功能需求, 在完成该需求之后则对其进行标记, 当需要增加或者删除需求的时候, 便对其直接操作。

具体格式如下:

软件配置项	关联的 CSCI 需求
员工管理模块	员工信息录入、员工信息查询、员工信息更新、员工信息删除
部门管理模块	部门信息录入、部门信息查询、部门信息更新、部门信息删除
薪资管理模块	资信息录入、薪资信息查询、薪资信息更新
统计分析模块	数据汇总统计、数据图表展示

每个 CSCI 需求应该被追踪到实现它的一个或多个软件配置项。这种追踪可以通过需求管理工具或文档中的表格来记录。

CSCI 需求	分配的软件配置项
---------	----------

员工管理需求	员工管理模块
部门管理需求	部门管理模块
薪资管理需求	员工管理模块、薪资管理模块
统计分析需求	统计分析模块，员工管理模块，薪资管理模块

7 注解

背景信息:

本文档旨在定义软件系统的结构和关键组件，以便开发人员和利益相关者了解该系统的设计和

功能。

词汇表:

本文档中使用的专业术语和其对应的定义详见附表 A：词汇表。

主要原理:

本文档遵循软件工程的最佳实践，将系统设计细节化并与功能需求一一对应，以便实现一个

功能完善且可靠的软件系统。

术语和定义:

本文档中使用的缩略语及其含义见附录 A

必要信息:

在阅读本文档之前，读者应熟悉软件开发和设计的基本原则，具备相关的领域知识和技术背

景。

重要说明:

本文档中的所有设计和决策均应该在软件开发过程中经过充分讨论和验证，以确保系统能够

按照预期方式运行。

持续更新:

本文档内容可能随着项目的进展和需求的变化而更新，读者在阅读时应关注最新版本以获取

最准确的信息。

附录

附录可用来提供那些为便于文档维护而单独出版的信息(例如图表、分类数据)。为便于处理，

附录可单独装订成册。附录应按字母顺序(A, B 等)编排。

附录 A:

以下是在本文档中使用的缩略语及其含义:

CSCI - 软件配置项 (Computer Software Configuration Item)

SEC - 安全性 (Security)

DM - 数据模型 (Data Model)

UTL - 实用工具 (Utility)

SDD - 系统设计文档 (System Design Document)

GDPR - 通用数据保护条例 (General Data Protection Regulation)

ERM - 实体关系模型 (Entity-Relationship Model)

DDL - 数据定义语言 (Data Definition Language)

分工情况:

栗锦虹: 1.引言 , 4.1 , 4.2, 4.5, 6.需求可追踪性

刘喜: 3.CSCI 体系结构设计 , 4.1, 4.2, 4.3, 4.4 5.CSCI 详细设计

田凯峰: 各类图表

高子寒: 6.需求可追踪行, 4.1

李华焘: 5.CSCI 详细设计, 4.2